Systems Update

Components/Architecture/Practices

2016 GHRC User Working Group Meeting
Sept 20-21, 2016

NASA

itsc
INFORMATION TECHNOLOGY AND SYSTEMS CENTER

THE UNIVERSITY OF ALABAMA IN HUNTSVILLE

GHRC System Updates

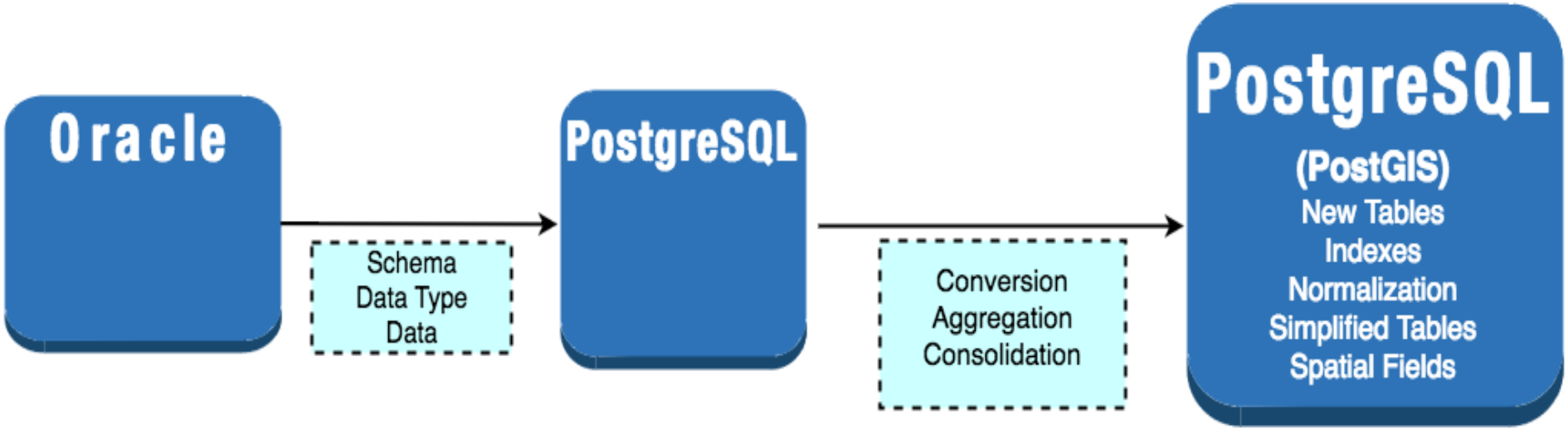
- Components
  - Database
  - Tools
- Architecture
  - Service Oriented Architecture (SOA)
- Practices
  - Development and deployment

Database

Database update

- Oracle to PostgreSQL/PostGIS
- Schema enhancements
  - Old database structure designed based on legacy hardware constraints
    - memory, processor speed, browser
  - Support hazard event model
- Benefits
  - Cost
  - Spatial support
  - Simplification
  - Flexibility

Transition approach

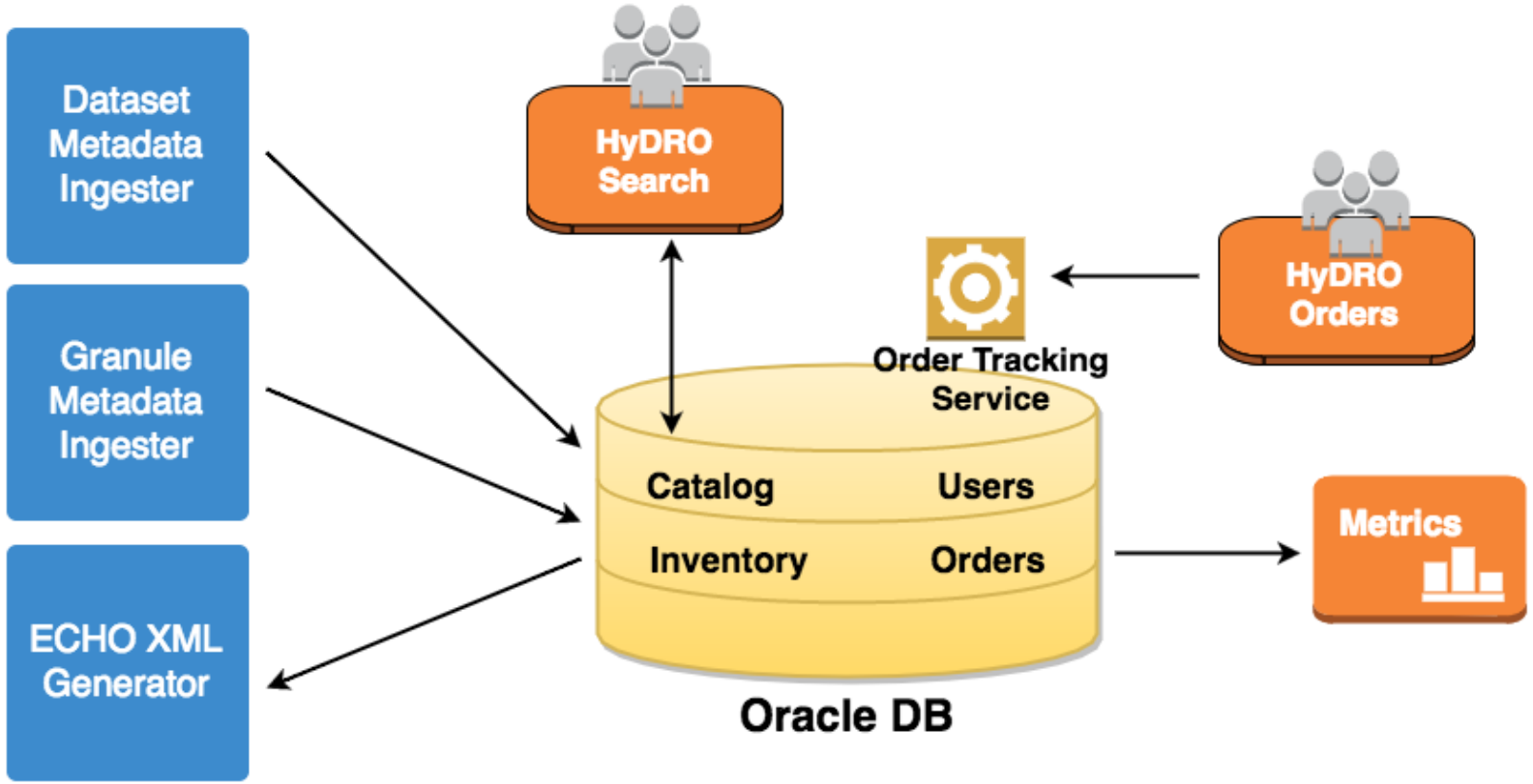


- Consolidate database schemas
- Normalize and simplify database schema
- Index for better performance
- Isolate database with service layer

**Production version completion by December 2016*

Architecture

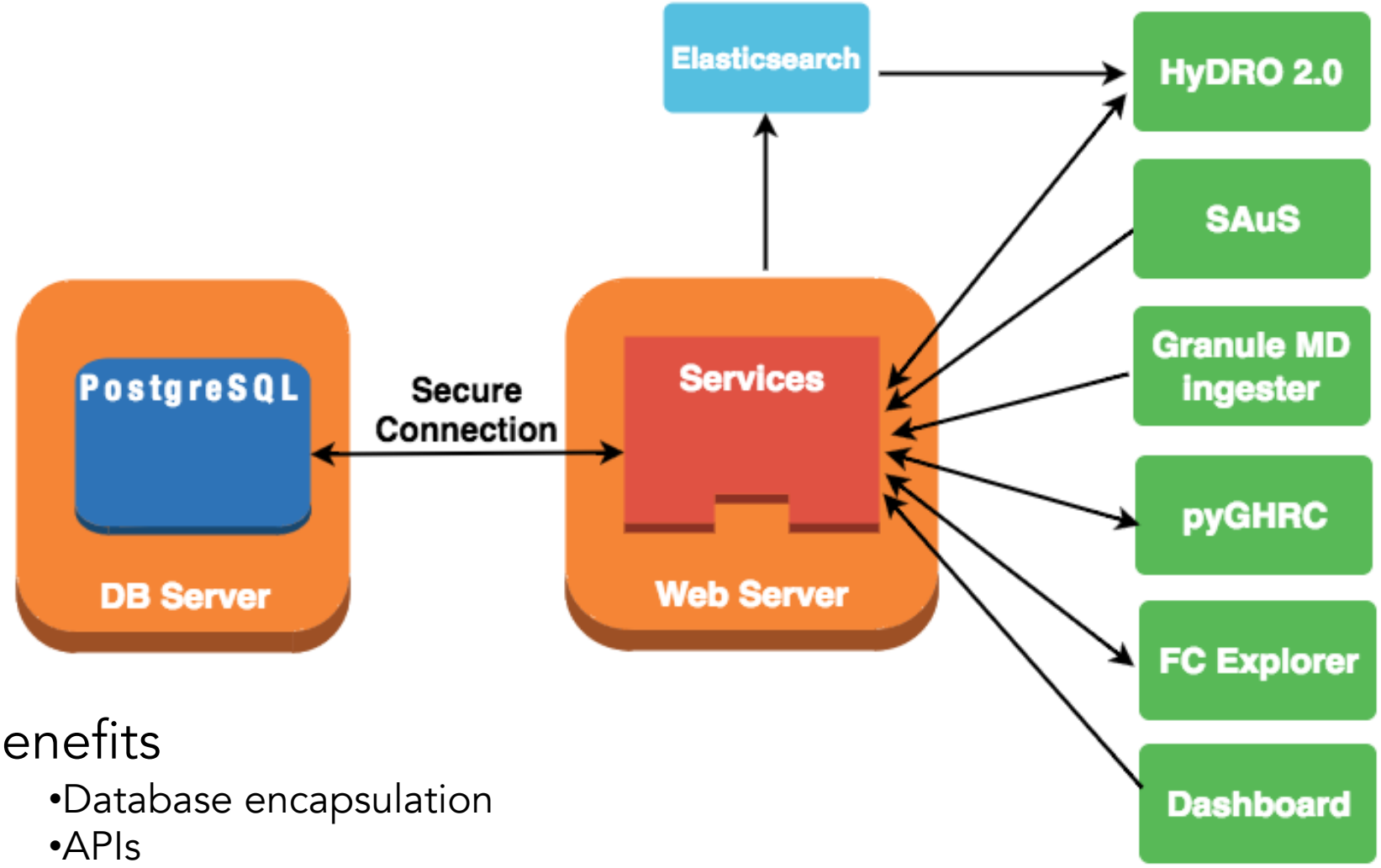
Legacy Processes



Drawbacks

- Tightly coupled
- Difficult to find clean integration points
- Rigid architecture makes even small changes complex
- Limited concept of API

Service Oriented Architecture



Benefits

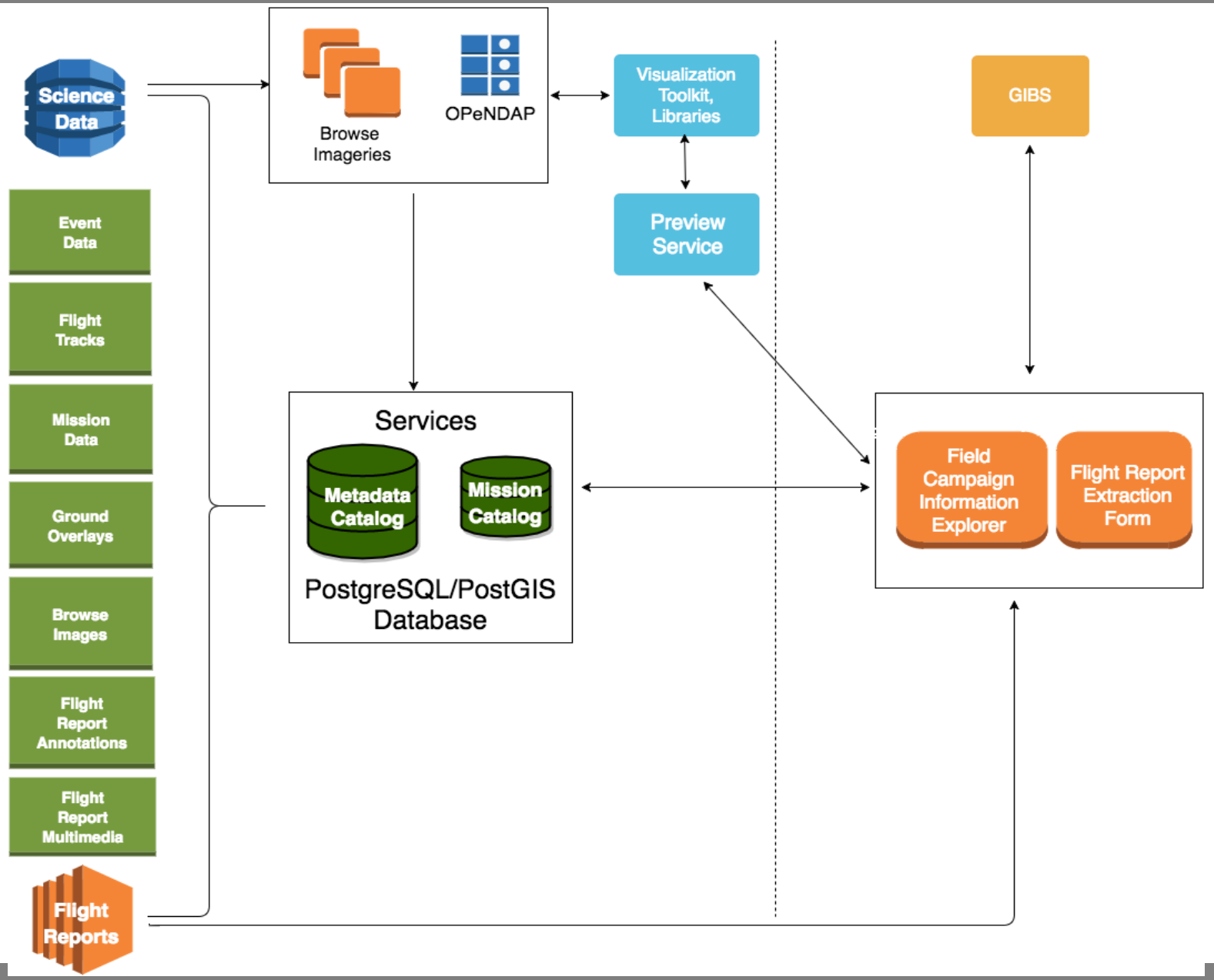
- Database encapsulation
- APIs
- Loosely coupled
- Easy integration for future tools and ESB
- Flexibility for future changes

Tools

API-driven (re-engineered) tools

- HyDRO 2.0
- FCX
- Metadata Catalog Ingest
- Metadata Export to CMR
- Elasticsearch
- Dashboard

Example: FCX with service APIs



DevOps

Practice that automates and closely monitors development and deployment

- Goals:
  - automation
  - rapid prototyping
  - support future improvements in developer collaboration
  - integration-testing
  - packaging and deployment
- Tools:
  - Amazon Web Services for computing
  - Gitlab – source code repository and automation
  - Jenkins – Continuous integration
  - Selenium – Automated web testing
  - Bamboo – Continuous integration
  - JIRA – Bug tracking and requirements

GHRC DevOps Workflow